CIRRELT

Centre interuniversitaire de recherche
sur les réseaux d'entreprise, la logistique et le transport

Interuniversity Research Centre
on Enterprise Networks, Logistics and Transportation

Location and Service Network Design

Teodor Gabriel Crainic
Mike Hewitt
Michel Toulouse
Duc Minh Vu

September 2015

CIRRELT-2015-45

Bureaux de Montréal :
Université de Montréal
Pavillon André-Aisenstadt
C.P. 6128, succursale Centre-ville
Montréal (Québec)
Canada H3C 3J7
Téléphone : 514 343-7575
Télécopie : 514 343-7121

Bureaux de Québec :
Université Laval
Pavillon Palasis-Prince
2325, de la Terrasse, bureau 2642
Québec (Québec)
Canada G1V 0A6
Téléphone : 418 656-2073
Télécopie : 418 656-2624

www.cirrelt.ca

Université Laval · McGill · Concordia · ÉTS · UQÀM · HEC Montréal · Polytechnique Montréal · Université de Montréal

Location and Service Network Design

Teodor Gabriel Crainic[1,2,*], Mike Hewitt[3], Michel Toulouse[1,4], Duc Minh Vu[3]

[1] Interuniversity Research Centre on Enterprise Networks, Logistics and Transportation (CIRRELT)

[2] Department of Management and Technology, Université du Québec à Montréal, P.O. Box 8888, Station Centre-Ville, Montréal, Canada H3C 3P8

[3] Department of Information Systems and Operations Management, Quinlan School of Business, Loyola University, 1 E. Pearson, Suite 204, Chicago, IL 60611, USA

[4] Vietnamese-German University, Le Lai Street, Hoa Phu Ward, Binh Duong New City, Binh Duong Province, Vietnam

Abstract. We present a new planning model for freight consolidation carriers, one that links strategic, resource-acquisition and allocation decisions with tactical, service network design related decisions. Specifically, like service network design models that recognize resource constraints, the model selects services and routes both commodities and the resources needed to support the services that transport them. However, the model also makes strategic decisions such as how many resources should be acquired, to what terminal new resources should be assigned, and which existing terminal-based resources should be reassigned. As such, the model can be used from a strategic planning, resource-acquisition and allocation perspective as it provides an estimate of the impact of such decisions on transportation costs. We adapt a solution approach for a service network design problem with a fixed set (both in number and allocation) of resources to one that can also make these resource acquisition and allocation decisions. Then, with an extensive computational study, we demonstrate the efficacy of the approach and benchmark its performance against both a leading commercial solver and a column generation-based heuristic. Finally, we perform an extensive computational study to understand how the resource-related and service network design-related components of the model interact, including how freight volumes and cost structures impact how many resources should be acquired.

Keywords: Service network design, fleet sizing and management, matheuristics, slope scaling, column generation.

Acknowledgements. While working on this project, the fourth author was doctoral student with Computer Science and Operations Research Department, Université de Montréal, and student member of CIRRELT. Partial funding for this project has been provided by the Natural Sciences and Engineering Research Council of Canada (NSERC), through its Discovery Grant program and by the EMME/2-STAN Royalty Research Funds (Dr. Heinz Spiess). We also gratefully acknowledge the support of Fonds de recherche du Québec – Nature et technologie (FRQNT) through their infrastructure grants and of Calcul Québec and Compute Canada through access to their high-performance computing infrastructure.

Results and views expressed in this publication are the sole responsibility of the authors and do not necessarily reflect those of CIRRELT.

Les résultats et opinions contenus dans cette publication ne reflètent pas nécessairement la position du CIRRELT et n'engagent pas sa responsabilité.

---

* Corresponding author: TeodorGabriel.Crainic@cirrelt.ca

Dépôt légal – Bibliothèque et Archives nationales du Québec
Bibliothèque et Archives Canada, 2015

© Crainic, Hewitt, Toulouse, Minh Vu and CIRRELT, 2015

1 Introduction

Consolidation carrier is an umbrella term for transportation companies that specialize in customer shipments that are too small, relative to vehicle capacity, or have too low a value, or both, to be economical to transport them directly from customer origin to customer destination. One finds in this category a huge part of the transportation industry, including small-package transportation companies, Less-than-Truckload (LTL) motor carriers, railroads, ocean/maritime liner navigation, river barge navigation, and the national and international container-based intermodal transportation. Shipments are then sorted and consolidated into vehicles, trucks, trailers, rail cars, etc., or boxes, containers generally, these being often further consolidated into larger vehicles or convoys, e.g., containers are consolidated on ships, packages into rail cars, and rail cars into blocks and trains.

These industries are large. To illustrate, in the U.S.A. alone, LTL freight is a roughly $30 billion industry and small package is even larger, with one player alone (UPS) reporting $54 billion in revenue in 2012. Carriers in both industries run high volume operations, with LTL carriers often spending millions of dollars in transportation and handling costs each week, and UPS delivering 16.3 million packages and documents a day in 2012.

Consolidation carriers support for a large part international trade and the supply chains, both domestic and global, that enable product manufacturers to provide these products in a timely and low-cost manner to manufacturing/assembly industrial firms, to wholesales and distribution/retail companies, and to final consumers. The LTL and small package carriers play a prominent role in the fulfillment of product orders placed online, in brick-and-mortar stores, and through other channels. The resources (power units, vehicles, manpower) that a carrier employs, both how many of each type they have available for use and what each resource's region or territory is, greatly impact the costs the carrier will incur when providing transportation services. In this paper, we present a model, and computationally effective solution method, that will help planners make strategic decisions regarding resource acquisition and allocation with a clear estimate of the impact of those decisions on transportation costs.

Consolidation carriers reduce costs through economies of scale. To achieve this, carriers route shipments through a network of terminals wherein they consolidate shipments from various customers and with various origins and destinations into the same vehicles. Transportation between two terminals in the network, with or without intermediate stops, is often referred to as a *service*. The primary cost associated with a service is the fixed cost incurred by hauling vehicle and transporting shipments and. thus. economies of scale correspond to services that move vehicles that are (nearly) full. At the same time, routing shipments through this network of services and consolidation terminals increases the transportation time (and distance) over transporting them directly from origin to destination. As such, consolidation carriers must balance customer demands for shorter

and shorter shipping times with their pursuit of economies of scale. The planning processes of carriers have long been assisted by solving the Service Network Design (SND) problem (Crainic, 2000), which determines the services to execute to support the timely delivery of customer shipments and the path each shipment should follow.

To execute a service, carriers need resources, including equipment and manpower. For example, while automation is becoming more and more prevalent in terminals, human resources are still used to unload shipments from inbound containers and handle and sort shipments to load into outbound containers. And resources are needed to provide the transportation itself. All modes require some type of power unit (trucks in trucking, locomotive engines in rail, planes in air, and vessels in maritime), an operator (driver, rail train operator, pilot, and captain), and sometimes a whole crew (particularly in air and sea). And of course, the vehicles and containers that hold shipments while in transit represent another resource.

These resources are often scarce, either because they are expensive to acquire (many power units), or there are few available for acquisition. The latter case is particulary true for operators. Some (Lewinski, 2014) have observed that there is or will soon be a shortage in airline pilots in the United States (primarily for air travel but likely to also impact air cargo eventually). Similarly, driver shortages have long plagued (Schulz, 2015) the trucking industry. While truckload carriers (those that transport customer shipments directly from their origins to their destinations) have been hardest hit by the driver shortage, often experiencing turnover rates that exceed 100%, LTL carriers have also struggled to find sufficient driver capacity. Compounding the lack of operators (pilots, drivers) today is the demographics associated with these professions; many individuals in these jobs at or near retirement age (particularly pilots wherein retirement is mandated at 65 years of age in the United States). Similarly, many rail carriers are now reporting that they do not have (and can not acquire) enough rail cars to meet the transportation needs of their customers (Trudell, 2015; Tita, 2014). This may be particularly problematic when, e.g., intermodal rail, operators must decide every year on the number, type and home region of the vehicles they lease.

There are often rules governing the use of these resources. Power units must be periodically maintained. Operators and crews can only work (fly, drive) a certain number of hours before they must take a rest. And some rules relate to the resource's "home" location or terminal, such as a driver needing to return home periodically (such as every day or every week). Similarly, the compensation for human resources (particularly drivers), either in terms of pay rates or signing bonuses may differ by the individual's home location. As such, a carrier must decide not only how many resources to acquire but also where those resources should be allocated or what regions should be targeted in recruiting efforts. Of course, requiring a resource to periodically return to its home terminal also impacts the services it can support.

Carriers sometimes move a resource (particularly those that support transportation) not to support a service but to position it for the execution of a future service. Such "empty" moves are common enough for many consolidation carriers that they can represent a significant cost. As a result, much of the recent work in Service Network Design also models the need to re-position resources (sometimes called "assets"). However, much of this research assumes an unlimited number of resources and does not model that an individual resource may need to periodically return to its home terminal. More recent research (Crainic et al., 2014) models a fixed and finite set of resources, one of which is needed for each service to execute, with each resource having a home terminal to which it must return periodically. For example, the model presented there could be used to account for drivers in a trucking setting, with each driver having a home terminal, or "domicile" to which he/she must return after a period of time dictated by the operational setting. In some settings, drivers may return home once a week. In other settings, a driver may return home every day. That research assumed the total number of resources and the assignment of each resource to a home terminal had been determined *a priori*. Also, the model only accounted for one type of resource.

As such, one extension of the research presented in Crainic et al. (2014) is to model the use of multiple types of resources, each with its own set of rules dictating what it can and cannot do. Some trucking companies have multiple classes of drivers, with one class representing those that return to their domicile weekly and another representing those which must return daily. Similarly, a transportation company may wish to use resources of different ages differently. For example, a trucking company may wish to use newer, more fuel-efficient trucks on its longer hauls and older, less fuel-efficient trucks to support shorter services. Another extension, and what we focus on in this paper, is to expand the scope of the decisions made by the model to include strategic, fleet acquisition and management-type choices.

Specifically, we extend the model presented in Crainic et al. (2014) to represent the following actions: the acquisition of new resources, including their assignment to a home terminal; and the reassignment of an existing resource's home terminal. Thus, the model we present has two layers: (1) a strategic layer, wherein resource acquisition and allocation decisions are made, and, (2) a tactical layer wherein service network design decisions are made along with decisions regarding the management of resources used to execute services. This second, tactical, layer serves as an estimator to the strategic layer of the impact of its decisions on transportation costs. We also include in the tactical layer the option of using a third party's resource (at extra cost) to support the transportation associated with a service. As these are third party resources, we do not model that the carrier must return them to a home terminal; instead we assume they are available to execute the service and no further management is needed.

We study this problem wherein both strategic fleet management and tactical transportation planning decisions are made in two ways: (1) we adapt the algorithm presented

in Crainic et al. (2014) to this new problem and computationally study its sensitivity to problem parameters, and, (2) we perform an extensive computational study to understand how the resource-related and service network design-related components of the model interact, including how freight volumes and cost structures impact how many resources should be acquired.

The paper is organized as follows. We begin with a detailed description of the problem in Section 2, after which we review the relevant literature to this problem in Section 3. Next, we detail the network model and mathematical formulation in Section 4. Section 5 is dedicated to the proposed matheuristic solution approach, while Section 6 presents the experimental study. We conclude in Section 7.

2 Problem Statement

We consider a problem that encompasses both strategic and tactical planning decisions for a consolidation-based carrier. We assume the carrier transports freight through a network of terminals on services. The tactical decisions select the services to operate an determine how freight is routed through this service and terminal network, with these services and routes supported by resources which are assigned to terminals. The strategic decisions determine the total number of resources available and the assignment of each resource to a home terminal. Costs associated with these strategic decisions include the purchase cost of a capital asset, the additional salary and/or signing bonus associated with hiring an individual, and transportation costs associated with re-allocating a resource from one home terminal to another. These strategic decisions can have a profound impact on the cost of transporting freight. While these decisions can be (and often are) made independently, we propose to solve them jointly; in this way the strategic, resource acquisition and allocation decisions can be made with an accurate estimate of their impact on the transportation costs the carrier will incur.

This estimate will be based on the costs associated with operating services and consolidation terminals to transport customer demands during a representative period of time, which we refer to as the *tactical planning horizon* (e.g., six months to one year). During this period of time, the carrier must transport a set of known (forecast) customer shipments; associated with each shipment is an origin location and availability time, a destination location and due arrival time, and a size. At terminals, shipments are sorted and consolidated into vehicles (and convoys, eventually) that will thus contain shipments from multiple customers, with different shipments potentially having different origins or destinations or both. Direct services (no intermediate stops) connect these terminals and specify how vehicles move. A *service* is thus defined by an origin terminal and a destination terminal, as well as by temporal characteristics: the time window during which the service will depart from the origin terminal and the time window during which it

will arrive at the destination terminal. Resources, which are assigned to terminals, are associated to services and provide the means to perform them. Each resource operates according to cycling routes, supporting a sequence of services, starting and returning to its assigned terminal.

Shipments will thus be routed through the service network, being sorted and consolidated at each intermediary terminal on this route. Shipment routing also displays a temporal component as the carrier has to decide when to start it. Indeed, shipments may be held at a terminal for a later-departing service so that it may be possible to perform a better consolidation with later-arriving shipments. Of course the decision to hold a shipment to achieve greater consolidation must be balanced against the need to deliver the shipment at the time the customer expects. There are various costs associated with executing a service, including costs associated with terminal operations that support the service and the transportation itself. Similarly, there are costs associated with handling a shipment at a terminal. We will discuss these in greater detail in the next section when we present the model.

As it is usual for tactical planning and service network design models, we focus on the regular and repetitive part of demand and operations. More precisely, it is assumed that shipment demands during the tactical planning horizon are representative and display a regular, repetitive pattern. Based on history, relations with customers and company operating policy, the planner selects a certain period, a week, say, to be representative of demand and operations. It is then assumed that the carrier will see the same demand repeated each week for the duration of the tactical planning horizon. Services and resource routes will therefore be selected and scheduled for the same time length, called *schedule length*.

This work extends the research presented in Crainic et al. (2014), which models that a single resource is needed to execute a service and that these resources are assigned to terminals to which they must periodically return. We only presume that the execution of a service requires a resource, not that a shipment needs to be assigned to a single resource for transportation from its origin to its destination. As such, a shipment may be transferred from one resource to another, and in the extreme case a shipment may travel on a sequence of services, with each of those services supported by a different resource. The rules governing the movements a resource may make during the schedule length can be complex and depend upon what the resource is; for a human resource (say a driver) government agencies (such as the United States Department of Transportation) specify many limits (sometimes called "hours-of-service" regulations) upon what they may do (FMCSA, 2014). For example, a driver may drive at most 11 hours a day after 10 consecutive hours off duty (at rest).

We consider a simple set of rules governing what movements a resource may make in this paper, namely that the resource must return to its home terminal at least once during

the tactical planning horizon. However, the model and solution method we propose can be easily adapted to other cases. There are also costs, such as those incurred due to maintenance, associated with using a resource from a specific home terminal during the tactical planning horizon. The last resource-related decision we consider is the option that a service be supported not by a resource owned (or leased) by the transportation company, but instead by a third party. In this situation, the resource is "acquired" from the third party only for the execution of this service and the carrier itself need not ensure that the resource's complete schedule (which may include moves for other carriers) follows the appropriate rules. Outsourcing a service to a third party-owned resource incurs costs that are greater than executing the service with an owned resource.

Ultimately, the problem the planner faces is to determine the number and allocation of resources that best balances the extra costs associated with resource acquisition and re-allocation with the savings these decisions enable in terms of transportation. Of course balancing these costs also necessitates putting them on the same scale; for a schedule length of a week (or even a month) the savings seen by purchasing a new truck will rarely outweigh its purchase cost. As such, when defining this problem, we assume that the acquisition and re-allocation costs are amortized or spread out over a series of periods of time that equal the length of the schedule length. For example, suppose we are considering a resource that costs \$200,000 to acquire, is expected to have a lifespan of ten years, and will be in use 50 weeks a year. With a schedule length that is one week long, we will amortize the acquisition cost to be \$200,000/(50*10) = \$400. We next review the literature relevant to this problem.

3 Literature Review

The planning problem we study connects two types of decisions: determining the acquisition and allocation of resources and how to transport customer shipments using those resources. The resource acquisition and allocation decisions can be seen as facility location-type decisions, whereas determining how to transport customer shipments can be viewed as service network design-type decisions. As such, we next review related literature in the facility location and service network design domain. At the same time our problem locates resources, not facilities, so we conclude this section with a review of the literature on service network design problems that recognize the need for resources and how they must be managed.

We first refer the reader to the review of Contreras and Fernandez (2012), which provides a unified view of problems that combine location and network design issues. Melkote and Daskin (2001b) present an optimization model that both chooses locations for (uncapacitated) facilities and designs a transportation network. This transportation network is used to route each customer shipment from its origin to the nearest avail-

able facility. The objective of the model is to minimize the combined cost of opening facilities, operating services and routing shipments. They transform the problem into an uncapacitated fixed cost network design problem that can be solved in reasonable run-times using a mixed integer programming solver. Building off this work, Melkote and Daskin (2001a) introduce a combined facility location/capacitated network design problem in which facilities have capacities on the amount of customer shipment demand they can serve. However, neither model captures the resources that are needed to support the transportation network.

In short, and to the best of our knowledge, resource acquisition, allocation, and management decisions have not been considered in the literature on location and network design problems. As such, we next turn our attention to the literature on service network design problems that recognize the need for and management of resources.

Early papers (Kim et al., 1999; Smilowitz et al., 2003; Lai and Lo, 2004) studied problems that modeled the requirement that the number of services entering and leaving a terminal at a point in time must be equal. These models assume one type of resource and that each service is supported by one unit of that resource. As a result, this constraint (often called the "design-balance" constraint) ensures a balance of resources at each terminal and point in time. Similar types of constraints can be found in papers wherein the resource modeled is a container (Powell, 1986; Jarrah et al., 2009; Erera et al., 2013).

Researchers (Pedersen et al., 2009) have observed that the addition of these design-balance constraints can complicate the search for high quality solutions as rounding-based techniques are likely to produce an infeasible solution. As a result, Pedersen et al. (2009) proposed a two-phase tabu-search method wherein the first phase explores the space of solutions that satisfy flow constraints but not necessarily design-balance constraints. The second phase is entered when a solution from the first does not satisfy the design-balance constraints, wherein a path-based neighborhood heuristic is used to convert the solution to one that is feasible for the full problem. However, the quality of the solution depends heavily on this second phase, which they observed required a significant number of iterations to produce a feasible solution.

Following up on that work, Vu et al. (2013) proposed an approach that can efficiently convert an infeasible solution (which satisfies the flow constraints but not design-balance constraints) to a feasible one using a minimum cost maximum flow procedure. The procedure is integrated into a three-phase matheuristic which combines tabu-search, path-relinking and exact optimization and this solution approach was found to be effective at finding high-quality solutions in reasonable run-times. In addition, this minimum cost maximum flow model was also used (Crainic et al., 2014) in a solution method for another service network design problem that models resource constraints and was effective in that setting as well. Simultaneously, Chouman and Crainic (2015) proposed a competitive matheuristic based on cutting plane approach which was able to produce high quality

solutions in short running times.

The design-balance constraints naturally imply a cycle-based formulation. As such, Andersen et al. (2009) compared cycle and arc-based formulations and observed that the use of cycle-based formulations enabled a more effective search for high quality primal solutions and yielded stronger dual bounds. As a result, Andersen et al. (2011) presented a cycle-based branch-and-price solution method to solve this problem for moderate instance dimensions.

However, these cycle-based formulations were used not as a modeling tool but because of their impact on algorithmic effectiveness. Crainic et al. (2014) instead used a cycle-based formulation to model a limit on how many resources are available at each terminal and that there are rules regarding what a resource may do during the planning horizon. The authors present a solution approach for this problem that combines column generation, slope-scaling, and exact optimization and with an extensive computational study illustrating its effectiveness.

4 Optimization model

We first discuss how we model the decisions made during the tactical planning horizon. During this period we assume the carrier transports shipments through a physical network of terminals, represented by the set Λ; for simplicity of presentation, and without loss of generality, we do not represent the infrastructure, roads or rail tracks, over which transportation is performed. Services are used to transport shipments. Σ stands for set of potential direct services, among which the model will select and schedule those to be included into the plan.

Specifically, at the tactical level of planning, services are selected and scheduled over a number of periods making up the *schedule length*. We assume this schedule length is divided into $\mathcal{T} = \{1, 2, \ldots, TMAX\}$ time periods. The selected plan will then be repeated on a schedule-length basis.

We model the operations of a carrier with a time-space network, $\mathcal{G} = (\mathcal{N}, \mathcal{A})$, where terminal activities in different periods are modeled with different nodes. Specifically, we use the node set $\mathcal{N}$ to model the operations of terminals in different periods, i.e., $\mathcal{N} = \Lambda \times \mathcal{T} = \{l_t | l \in \Lambda, t \in \mathcal{T}\}$, where l_t represents terminal l at period t. The arc set $\mathcal{A}$ contains two types of arcs. The first is a *service arc* (from the set Σ) and models the operation of a service between two terminals at a particular point in time. The second is a *holding arc* and models the opportunity for a resource or shipment to idle at a terminal from one period to the next. We denote the set of service arcs by $\mathcal{S}$ and holding arcs by $\mathcal{H}$ and thus $\mathcal{A} = \mathcal{S} \cup \mathcal{H}$.

Regarding service arcs, for each possible service $s = (l, m)$ between terminals $l, m \in \Lambda$ and time period $t \in \{1, \ldots, TMAX\}$, we add the arc $(l_t, m_{t'=(t+\pi) \bmod TMAX})$ to $\mathcal{S}$ (assuming the arc requires π periods of travel time). Due to our presumption that freight demands follow a repetitive pattern, and thus the schedule our model prescribes will be repeated, our time-space network "wraps around." Specifically, we model a service of length π that departs from a terminal in period t as arriving at the destination in period $(t + \pi)$ mod $TMAX$. We also assume a limit, u_s, on how much shipment demand can be carried by service $s = (l, m)$ and set the capacity, $u_{l_t m_{t'}}$, of executions of that service at different times t to u_s. Regarding the *holding arcs*, we add to $\mathcal{H}$ arcs of the form $(l_t, l_{(t+1) \bmod TMAX})$ for each terminal l and period t. We create holding arcs that "wrap around" as well. While we assume these arcs are uncapacitated (both with respect to shipment demands and resources) in our experiments, terminal capacities (on shipments or resources) could be modeled by placing capacities on these arcs.

We model a shipment that needs to be delivered from terminal l and is available in period t to terminal, m, by period t' as a commodity with index k, origin node $o(k) = l_t$, and destination node $d(k) = m_{t'}$. We denote the size of this shipment as w_k. The set of all shipments is represented by $\mathcal{K}$. The model prescribes the routing from $o(k)$ to $d(k)$ for commodity k through the continuous and non-negative variable $x^k_{l_t, m_{t'}}$. Such variables are defined for each $(l_t, m_{t'}) \in \mathcal{A}$ and represent the portion of demand associated with commodity k that travels on arc $(l_t, m_{t'})$, which may be a service or a holding arc. We associate the variable cost $c^k_{l_t m_{t'}}$ with commodity k traveling on arc $(l_t, m_{t'}) \in \mathcal{A}$. For a service arc, this parameter can model handling costs associated with loading the shipment into a vehicle at the origin terminal and unloading at the destination terminal. This parameter can also model the impact the weight of a shipment can have on the cost of executing a service. For a holding arc, this parameter can model other handling activities, or, the allocation of the cost of physical space to shipments based on the amount of space in the terminal they require.

For a commodity to travel on the service arc $(l_t, m_{t'}) \in \mathcal{S} \subset \mathcal{A}$, that service must be "executed", and doing so requires the use of a resource which must periodically return to its assigned home terminal. Similar to the research presented in Crainic et al. (2014), we model a sequence of possible movements during the schedule length for a resource assigned to terminal l with a cycle in the graph $\mathcal{G}$ that begins and ends at node $l_t \in \mathcal{N}$ for some $t \in \mathcal{T}$. Hereafter, we refer to such a sequence of moves as an *itinerary*. We denote the set of such cycles by θ_{l_t} and set $\theta_l = \cup_{t=1}^{TMAX} \theta_{l_t}$, as the set of all cycles requiring a resource that is assigned to terminal l and depart from there at some time period during the schedule length. Thus, choosing cycle τ from θ_l implies specifying the itinerary for a resource assigned to terminal l and we model this choice with the binary variable z_τ. We associate the fixed cost F_l with the binary variable z_τ wherein $\tau \in \theta_l$, or, in other words, the use of a resource whose home terminal is l. We define $\theta = \cup_{l \in \mathcal{L}} \theta_l$.

We let the attribute $r^\tau_{l_t m_{t'}}$ (binary) denote whether arc $(l_t, m_{t'}) \in \mathcal{A}$ is contained in

cycle τ. We associate a fixed cost, f_{lm}, with a carrier executing a service departing from terminal l and arriving at terminal m with its own resources. This parameter can be used to model the actual transportation cost of a resource traveling from terminal l to terminal m. It can also be used to model other overhead costs associated with executing a service, such as facility maintenance and labor. This parameter can also be indexed by the time period t for settings wherein transportation costs are time-dependent, such as in areas where congestion-based traffic pricing is used. As such, the expression $f_{lm} \sum_{\tau \in \theta} r^{\tau}_{l_t m_{t'}} z_\tau$ represents the fixed cost paid for executing service (l, m) departing in period t. We also model the option of executing that same service, albeit with the use of a third party-owned resource. We model this choice with the binary variable $y^e_{l_t m_{t'}}$ and associate with it the fixed cost f^e_{lm}. For most practical settings, we anticipate this parameter value will be a function of the same overhead costs as those that contribute to the value of f_{lm}, but that the transportation cost associated with executing the service with a third party carrier will be higher.

The rules governing the movements a resource may make during the schedule length are encoded in the definition of the set θ_{l_t}. Note that this allows us to model rules that vary both by the terminal l to which the resource is assigned and the period t during which the itinerary begins. For our experiments, we only impose the rule that the itinerary for a resource must begin and end at the resource's assigned terminal. Thus a valid cycle is one that begins by departing from l in period t and ends by arriving at l in period t, albeit $TMAX$ periods later. We note that we allow a cycle beginning at l_t to return to l multiple times, and if it last returns to l in period $t' < t + TMAX$ we append holding arcs so that it reaches l_{t+TMAX}.

We next turn our attention to how we model the strategic decisions the planner must make. Conceptually (and, as we will see later, algorithmically) we model these choices in conjunction with the tactical planning decisions discussed above in the manner depicted in Figure 1. Namely, we add to the time-space network on which service choice and commodity transportation decisions are modeled a layer that models the choice of resource acquisition and allocation decisions. There are two types of nodes in this layer. The first is an "Acquisition node," which we denote by A, and represents the acquisition of a new resource. We connect this node to each of the terminals l at the beginning of the tactical planning horizon with an arc that represents the allocation of a newly acquired resource to that terminal. The second type of node is used to model the re-allocation of existing resources. As such, we add a node for each terminal $l' \in \Lambda$ to this layer and then arcs connecting that node to each terminal, $l \in \Lambda$ at the beginning of the tactical planning horizon. These arcs represent the re-allocation of a resource assigned to terminal l to terminal l'. We let Λ^+ denote the set of terminals, Λ, along with the Acquisition node.

We model both types of decisions with the variable $h_{ll'}$ with which we associate the cost parameter $H_{ll'}$. When l corresponds to the Acquisition node, the variable represents

the purchase of a new resource and subsequent allocation to terminal l'. As such, if the resource being modeled is equipment, H_W could model the acquisition cost, only amortized. If the resource is an individual this parameter could model wages and some amortization of a signing bonus paid to the individual. When l represents an existing terminal then the variable h_W corresponds to the allocation of a resource that is currently assigned to terminal l to terminal l'. When $l \neq l'$, this variable represents re-allocation and H_W represents any costs associated with such an action, such as transportation. When $l = l'$ this variable represents leaving resources at their currently assigned terminal, in which case we anticipate $H_W = 0$.. We let I_l represent the number of existing resources assigned to terminal l.

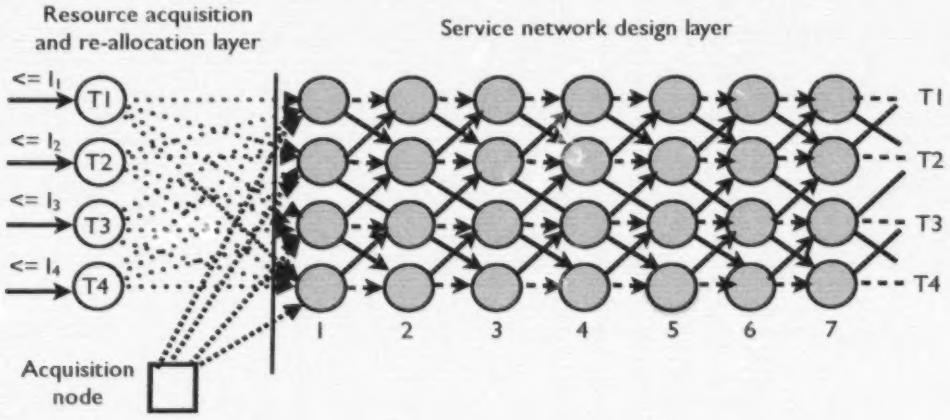

Figure 1: Network model of strategic and tactical decisions

Ultimately, we seek to solve what we call the Location with Service Network Design (LWSND) problem, which aims to

$$\text{minimize} \sum_{l\in\Lambda, l'\in\Lambda} H_W h_W +$$

$$\sum_{l\in\Lambda} F_l(\sum_{\tau\in\theta_l} z_\tau) + \sum_{(l_t,m_{t'})\in\mathcal{S}} f_{lm} \sum_{\tau\in\theta} r^\tau_{l_t m_{t'}} z_\tau + \sum_{(l_t,m_{t'})\in\mathcal{S}} f^e_{lm} y^e_{l_t m_{t'}} + \sum_{k\in\mathcal{K}} \sum_{(l_t,m_{t'})\in\mathcal{A}} c^k_{l_t m_{t'}} x^k_{l_t m_{t'}}$$

subject to

$$\sum_{l'\in\Lambda} h_W = I_l, \quad \forall l \in \Lambda, \tag{1}$$

$$\sum_{\tau\in\theta_l} z_\tau \leq \sum_{l'\in\Lambda^+} h_{Wl}, \quad \forall l \in \Lambda, \tag{2}$$

$$\sum_{j:(l_t,m_{t'})\in\mathcal{A}} x^k_{l_t m_{t'}} - \sum_{j:(m_{t'},l_t)\in\mathcal{A}} x^k_{m_{t'} l_t} = \begin{cases} w^k & \text{if } i = o(k) \\ 0 & i \neq o(k),\ d(k) \\ -w^k & \text{if } i = d(k) \end{cases} \quad \forall i \in \mathcal{N},\ \forall k \in \mathcal{K}, \tag{3}$$

$$\sum_{k \in \mathcal{K}} x^k_{l_t m_{t'}} \leq u_{l_t m_{t'}} (\sum_{\tau \in \theta} r^\tau_{l_t m_{t'}} z_\tau + y^e_{l_t m_{t'}}), \quad \forall (l_t, m_{t'}) \in \mathcal{S}, \tag{4}$$

$$\sum_{\tau \in \theta} r^\tau_{l_t m_{t'}} z_\tau + y^e_{l_t m_{t'}} \leq 1, \quad \forall (l_t, m_{t'}) \in \mathcal{S}, \tag{5}$$

$$h_{lt'} \in \mathcal{Z} \quad \forall l \in \Lambda^+, l' \in \Lambda, \tag{6}$$

$$x^k_{l_t m_{t'}} \geq 0, \quad \forall (l_t, m_{t'}) \in \mathcal{A}, k \in \mathcal{K}, \tag{7}$$

$$z_\tau \in \{0, 1\}, \quad \forall \tau \in \theta, \tag{8}$$

$$y^e_{l_t m_{t'}} \in \{0, 1\}, \quad \forall (l_t, m_{t'}) \in \mathcal{S}. \tag{9}$$

The objective is to minimize the total cost of resource acquisition and allocation decisons and the decisions made to support transporting customer shipments. The first two constraints of the model correspond to the srategic decisions, with the first, Constraint (1), ensuring that the allocation of existing resources that are assigned to terminal l does not exceed the number (l_l) that are currently assigned to l. Constraint (2) links the strategic resource acquisition and allocation decisions that determine the number of resources available at terminal l with the tactical decision of how many resources assigned to terminal l to use to support services. Note the summation over Λ^+ in Constraint (2) enables the use of resources that are newly acquired.

Constraints (3), (4), and (5) are variants on constraints often seen in Service Network Design modles. Constraint (3) ensures that each commodity is routed from its origin node to its destination node. Constraint (4) ensures that commodities only travel on a service when it is supported by a resource (either carrier or externally-owned) and that the capacity for that service is not exceeded. Constraint (5) ensures that each service is executed at most once. Finally, Constraints (6), (7), (8), and (9) define the domains of the variables in this problem. There are other, valid, constraints for this problem, such as disaggregated versions of Constraint (4). However, as these are not the focus of this research we omit them from this model.

5 Solution Approach

There are two challenges associated with producing a high-quality solution to the LWSND. First, even for moderately-sized instances, the set of resource cycles, θ, is too large to be enumerated in a reasonable run-time. Thus, an effective solution procedure must first determine a set of cycles that are likely to appear in a high-quality solution. And, when making this determination, the solution approach must also determine the appropriate source for each cycle. Second, given a set of cycles, the solution approach must extract a solution that satisfies the constraints of the problem and is of high quality. We provide

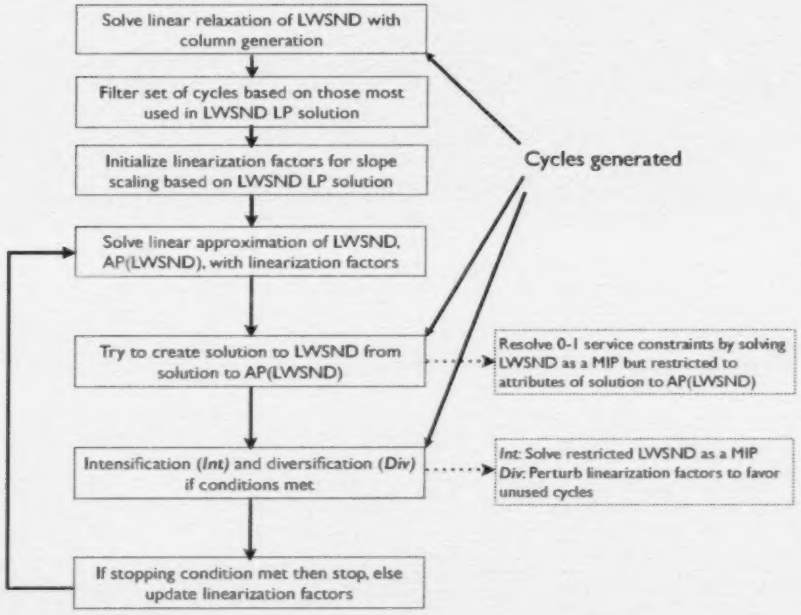
Solve linear relaxation of LWSND with column generation
Filter set of cycles based on those most used in LWSND LP solution
Initialize linearization factors for slope scaling based on LWSND LP solution
Solve linear approximation of LWSND, AP(LWSND), with linearization factors
Try to create solution to LWSND from solution to AP(LWSND)
Intensification (Int) and diversification (Div) if conditions met
If stopping condition met then stop, else update linearization factors
Cycles generated
Resolve 0-1 service constraints by solving LWSND as a MIP but restricted to attributes of solution to AP(LWSND)
Int: Solve restricted LWSND as a MIP
Div: Perturb linearization factors to favor unused cycles

Figure 2: Steps of solution approach

an overview of the steps of our proposed solution approach, and how they inter-relate, in Figure 2.

The solution approach we propose overcomes both of these challenges with techniques that were developed (Crainic et al., 2014) as part of a solution approach for the SNDRC, the tactical planning portion of the LWSND, wherein there are a fixed number of resources assigned to each terminal. More specifically, that solution approach explicitly handles that there is a limit on how many resource cycles from each terminal that can be selected. Regarding the first challenge, the solution procedure we propose uses column generation (Barnhart et al., 1998; Desaulniers et al., 2005) to determine the set of resource cycles that can be included in a solution. Of course, column generation is an algorithm for solving linear programs. As such, and regarding the second challenge, to produce a high-quality solution to the LWSND (an integer program), the approach we propose uses steps from both Slope Scaling (Crainic et al., 2004; Kim and Pardalos, 1999) and Matheuristics (Archetti et al., 2008; Chouman and Crainic, 2015; Vu et al., 2013; De Franceschi et al., 2006; Hewitt et al., 2010).

5.1 Column generation

We define LWSND($\bar{\theta}$) to be the LWSND restricted to the cycles in $\bar{\theta}$ and its linear relaxation to be the LWSND with the variables z_τ and $y^e_{l_t m_{t'}}$ allowed to take on fractional values. We note that due to the presence of the outsourcing variables, $y^e_{l_t m_{t'}}$, LWSND($\bar{\theta}$) is always feasible. As such, the linear relaxation of LWSND(θ) will be repeatedly solved, with new cycle variables, z_τ, added to $\bar{\theta}$ when reduced cost calculations indicate that they may lead to an improved LP solution. As in traditional column generation, these reduced cost calculations are based on dual variables associated with constraints that are binding at the current linear programming solution.

Specifically, we associate the dual variables $\gamma_l (\leq 0)$ with each constraint in set (2), $\alpha_{l_t m_{t'}} (\leq 0)$ with each constraint in set (4), and $\beta_{l_t m_{t'}} (\leq 0)$ with each constraint in set (5). As such, the reduced cost of variable $z_\tau, \tau \in \theta_l$ is given by the expression $F_l - \gamma_l - \sum_{(l_t, m_{t'}) \in \tau \cap \mathcal{S}} (f_{l_t m_{t'}} - \alpha_{l_t m_{t'}} u_{lm} - \beta_{l_t m_{t'}})$. Thus, given sets of dual variable values we form an acyclic graph wherein arcs are given the cost $f_{l_t m_{t'}} - \alpha_{l_t m_{t'}} u_{lm} - \beta_{l_t m_{t'}}$. We search for least-cost cycles in this graph with a dynamic programming approach similar to the one presented in Crainic et al. (2014). When the least-cost cycle has a negative reduced cost we add it to the set $\bar{\theta}$ and solve the LWSND($\bar{\theta}$) again. The procedure repeats until it reaches an iteration wherein no cycles with negative reduced cost are found. At this point the linear relaxation of the LWSND has been solved. As such, this procedure also yields a bound on the optimal value of the LWSND. However, we also use the cycles generated during this solution process and the solution to the linear relaxation itself to inform a slope scaling procedure that produces high-quality primal solutions. We next describe this slope-scaling procedure.

5.2 Slope scaling

Slope scaling (Kim and Pardalos, 1999; Crainic et al., 2004) is an algorithmic strategy for hard optimization problems that repeatedly solves approximations of the original problem, with these solutions then converted through a (usually simple) procedure to solutions to the original problem. These approximations are typically relaxations, and thus much easier to solve than the original problem, and are often defined by allowing integer variables to take on fractional values or removing other complicating constraints. The objective function of the approximation solved at an iteration is parameterized by *linearization factors*. Each time the approximation is solved and a solution to the original problem is constructed, the linearization factors are updated so that the cost of the solution to the approximation and the solution to the original problem agree. Consequently, defining a slope scaling procedure requires the definition of an approximation problem and a method for updating the linearization factors. We next discuss how we define each of these. We note that our slope scaling procedure begins with a fixed set

of cycles, $\bar{\theta}$, which we construct with a subset of the cycles found during the previously described column generation procedure. Specifically, we populate $\bar{\theta}$ with the cycles τ such that $z^*_\tau > \alpha$, wherein z^*_τ indicates the value of the variable in the optimal solution to the linear relaxation of the LWSND and α is an algorithm parameter.

When using slope scaling (Kim and Pardalos, 1999; Crainic et al., 2004) to produce solutions to an integer program (such as the LWSND), the approximation problem is typically the linear relaxation of the original problem. Our approximation problem, which we call AP(LWSND), is formed by both relaxing integrality constraints and removing other constraints, leaving what is essentially a multi-commodity flow problem. The approximation problem we define is similar to what was presented in Crainic et al. (2014) for the SNDRC, which linearized cycle variables z_τ. However, in the LWSND, we must also linearize the acquisition and allocation variables, $h_{ll'}$, and the outsourcing variables $y^e_{l_t m_{t'}}$.

To linearize these three variables (z_τ, $h_{ll'}$, and $y^e_{l_t m_{t'}}$), AP(LWSND) is defined with two flow variables. The first is $x^{k\tau}_{l_t m_{t'} ll'}$, which represents the flow of commodity k on service $(l_t, m_{t'})$ that is supported by resource cycle τ with the resource sourced from l. The second is $x^{ke}_{l_t m_{t'}}$, which represents the flow of commodity k on service $(l_t, m_{t'})$ that is supported by an outsourced resource. We note that we only define these new flow variables for arcs that are services as holding arcs do not require a resource. We relate these two flow variables to the flow variables of the LWSND, $x^k_{l_t m_{t'}}$ with the equation

$$\sum_{l \in \Lambda^+} \sum_{l' \in \Lambda} \sum_{\tau \in \bar{\theta}_{l'}} r^\tau_{l_t m_{t'}} x^{k\tau}_{l_t m_{t'} ll'} + x^{ke}_{l_t m_{t'}} = x^k_{l_t m_{t'}}.$$

Recall that after solving the approximation problem, the values of the variables ($x^{k\tau}_{l_t m_{t'} ll'}, x^{ke}_{l_t m_{t'}}$) are used to create a solution to the LWSND. To approximate the cost of this resulting solution in the AP(LWSND), we define linearization factors for each type of flow variable. Specifically, we associate the linearization factor $\rho^{k\tau}_{l_t m_{t'} ll'}$ with variable $x^{k\tau}_{l_t m_{t'} ll'}$ and $\phi^{ke}_{l_t m_{t'}}$ with variable $x^{ke}_{l_t m_{t'}}$. The purpose of these linearization factors is for the expression $\sum_{(l_t, m_{t'}) \in \mathcal{S}} \sum_{k \in \mathcal{K}} \rho^{k\tau}_{l_t m_{t'} ll'} x^{k\tau}_{l_t m_{t'} ll'}$ to approximate the cost $H_{ll'} z^\tau_{ll'} + F_{l'} z^\tau_{ll'} + \sum_{(l_t, m_{t'}) \in \mathcal{S}} f_{lm} \sum_{\tau \in \theta} r^\tau_{l_t m_{t'}} z_\tau$ of the resulting solution to the LWSND. Similarly, the expression $\sum_{k \in \mathcal{K}} \phi^{ke}_{l_t m_{t'}} x^{ke}_{l_t m_{t'}}$ should approximate the cost $f^e_{lm} y^e_{(l_t, m_{t'})}$ associated with the resulting solution. Formally, the AP(LWSND) is the following optimization problem:

$$\text{minimize} \sum_{l \in \Lambda^+} \sum_{l' \in \Lambda} \sum_{\tau \in \bar{\theta}_{l'}} \sum_{(l_t, m_{t'}) \in \mathcal{S}} \sum_{k \in \mathcal{K}} \rho^{k\tau}_{l_t m_{t'} ll'} x^{k\tau}_{l_t m_{t'} ll'} + \sum_{k \in \mathcal{K}} \sum_{(l_t, m_{t'}) \in \mathcal{S}} \phi^{ke}_{l_t m_{t'}} x^{ke}_{l_t m_{t'}}$$

subject to

$$\sum_{l \in \Lambda^+} \sum_{l' \in \Lambda} \sum_{\tau \in \bar{\theta}_{l'}} r^\tau_{l_t m_{t'}} x^{k\tau}_{l_t m_{t'} ll'} + x^{ke}_{l_t m_{t'}} = x^k_{l_t m_{t'}}, \tag{10}$$

$$\sum_{j:(l_t,m_{t'})\in\mathcal{A}} x^k_{l_t m_{t'}} - \sum_{j:(m_{t'},l_t)\in\mathcal{A}} x^k_{m_{t'} l_t} = \begin{cases} w^k & \text{if } i = o(k) \\ 0 & i \neq o(k),\ d(k)\ \ \forall i \in \mathcal{N},\ \forall k \in \mathcal{K}, \\ -w^k & \text{if } i = d(k) \end{cases} \quad (11)$$

$$\sum_{k\in\mathcal{K}} x^k_{(l_t,m_{t'})} \leq u_{lm} \quad \forall (l_t, m_{t'}) \in \mathcal{S}, \quad (12)$$

$$x^{k\tau}_{l_t m_{t'} l'} \geq 0, \quad \forall (l_t, m_{t'}) \in \mathcal{S}, l \in \Lambda^+, l' \in \Lambda, k \in \mathcal{K}, \tau \in \bar{\theta}, \quad (13)$$

$$x^{ke}_{l_t m_{t'}} \geq 0, \quad \forall (l_t, m_{t'}) \in \mathcal{S}, k \in \mathcal{K}. \quad (14)$$

Having solved the AP(LWSND), we have two sets of flow values, $\tilde{x}^{k\tau}_{l_t m_{t'} l'}$, and $\tilde{x}^{ke}_{l_t m_{t'} l'}$. The first indicate which resource cycles are operated by owned resources and how those resources are sourced. The second indicates which services are supported by a third party resource. We use these values to create a solution to the LWSND with the following rules:

$$\tilde{z}^{\tau}_{ll'} = \begin{cases} 1 & \text{if } \sum_{(l_t,m_{t'})\in\tau\cap\mathcal{S}} \sum_{k\in\mathcal{K}} \tilde{x}^{k\tau}_{l_t m_{t'} l'} > 0 \\ 0 & \text{otherwise} \end{cases}, \forall \tau \in \bar{\theta}_{l'}, l \in \Lambda^+, l' \in \Lambda, \quad (15)$$

and

$$\tilde{y}^{e}_{(l_t,m_{t'})} = \begin{cases} 1 & \text{if } \sum_{k\in\mathcal{K}} \tilde{x}^{ke}_{l_t m_{t'}} > 0 \\ 0 & \text{otherwise} \end{cases}, \forall (l_t, m_{t'}) \in \mathcal{S}. \quad (16)$$

The first rule (15) states that resource cycle τ, which requires a resource assigned to terminal l' and sourced from terminal l will be used if there is commodity flow on any of the services it supports. The second rule (16) states that a service will be operated by a third party resource if it transports any commodity flow. However, we note tha these variable values $(\tilde{z}^{\tau}_{ll'}, \tilde{y}^{e}_{(l_t,m_{t'})})$ along with the flows, $\tilde{x}^{k}_{l_t m_{t'}}$, in the solution to the AP(LWSND) may not define a feasible solution to the LWSND as they may violate constraints from the sets (1, 2, 4, and 5). We will discuss how we address infeasible solutions in the next section. We next discuss how we update the linearization factors, $\rho^{k\tau}_{l_t m_{t'} l'}$ and $\phi^{ke}_{l_t m_{t'}}$.

We let $\rho^{k\tau}_{l_t m_{t'} l'}(t)$ and $\phi^{ke}_{l_t m_{t'}}(t)$ denote the value of the linearization factors in iteration t. We update the ρ factors to satisfy

$$\sum_{(l_t,m_{t'})\in\mathcal{S}} \sum_{k\in\mathcal{K}} \rho^{k\tau}_{l_t m_{t'} l'}(t+1) x^{k\tau}_{l_t m_{t'} l'} = H_{l'} \tilde{z}^{\tau}_{l'} + F_{l'} \tilde{z}^{\tau}_{l'} + \sum_{(l_t,m_{t'})\in\mathcal{S}\cap\theta} f_{lm} \tilde{z}^{\tau}_{ll'}$$

As such, $\forall \tau \in \bar{\theta}_{l'}, l \in \Lambda^+$, and $l' \in \Lambda$ we set

$$\rho^{k\tau}_{l_t m_{t'} l'}(t+1) = \begin{cases} \frac{H_{l'} + F_{l'} + \sum_{(l_t,m_{t'})\in\mathcal{S}\cap\tau} f_{lm}}{\sum_{(l_t,m_{t'})\in\tau\cap\mathcal{S}} \sum_{k\in\mathcal{K}} \sum_{(l_t,m_{t'})\in\tau\cap\mathcal{S}} \sum_{k\in\mathcal{K}} \tilde{x}^{k\tau}_{l_t m_{t'} l'}} & \text{if } \sum_{(l_t,m_{t'})\in\tau\cap\mathcal{S}} \sum_{k\in\mathcal{K}} \tilde{x}^{k\tau}_{l_t m_{t'} l'} > 0 \\ \rho^{k\tau}_{l_t m_{t'} l'}(t) & \text{otherwise} \end{cases} \quad (17)$$

Similarly, we update the ϕ factors to satisfy

$$\sum_{k \in K} \phi^{ke}_{l_t m_{t'}}(t+1)\tilde{x}^{ke}_{l_t m_{t'}} = f^e_{lm}\tilde{y}^e_{(l_t, m_{t'})}$$

As such, $\forall (l_t, m_{t'}) \in \mathcal{S}$ we set

$$\phi^{ke}_{l_t m_{t'}}(t+1) = \begin{cases} \frac{f^e_{lm}}{\sum_{k \in \mathcal{K}} \tilde{x}^{ke}_{l_t m_{t'}}} & \text{if } \sum_{k \in \mathcal{K}} \tilde{x}^{ke}_{l_t m_{t'}} > 0 \\ \phi^{ke}_{l_t m_{t'}}(t) & \text{otherwise} \end{cases} \tag{18}$$

Finally, we initialize the linearization factors depending on whether they appear in the solution to the linear relaxation of the LWSND found by the column generation procedure. We follow Crainic et al. (2014), who observed that it was best to initialize the linearization factors so as to favor the cycles that appear in the solution to the linear relaxation. Specifically, $\forall \tau \in \bar{\theta}_{l'}, l \in \Lambda^+$, and $l' \in \Lambda$ we set

$$\rho^{k\tau}_{l_t m_{t'} l l'}(0) = \begin{cases} 1 & \text{if } \tau \text{ appears in solution to linear relaxation of LWSND,} \\ \frac{H_{ll'} + F_{l'} + \sum_{(l_t, m_{t'}) \in S \cap \tau} f_{lm}}{\sum_{(l_t, m_{t'}) \in S \cap \tau} u_{lm}} & \text{otherwise} \end{cases} \tag{19}$$

and $\forall (l_t, m_{t'}) \in \mathcal{S}$ we set

$$\phi^{ke}_{l_t m_{t'}}(0) = \begin{cases} 1 & \text{if } y^e_{l_t m_{t'}} > 0 \text{ in solution to linear relaxation of LWSND,} \\ \frac{f^e_{lm}}{u_{lm}} & \text{otherwise.} \end{cases} \tag{20}$$

When the solution $(\tilde{x}, \tilde{z}, \tilde{y})$ to AP(LWSND) violates a constraint from one of sets (1, 2, 4, and 5), our approach next executes the following procedures to try and construct a feasible solution to LWSND from $(\tilde{x}, \tilde{z}, \tilde{y})$.

5.3 Creating a feasible solution to LWSND from a solution to AP(LWSND)

Because the AP(LWSND) does not include all the constraints that are present in the LWSND (constraint sets 1, 2, 4, and 5), and unlike most slope scaling approaches, rounding procedures are not sufficient to construct a feasible solution to the LWSND from a solution, $(\tilde{x}, \tilde{z}, \tilde{y})$, to the AP(LWSND). Instead we must execute another procedure to construct a feasible solution to the SNDRC. Our procedure, instead of modifying the solution to the AP(LWSND) directly, creates a subgraph, $\bar{\mathcal{G}}$ of $\mathcal{G}$ that contains the nodes

$\mathcal{N}$ and a subset, $\bar{\mathcal{A}}$, of the arcs, $\mathcal{A}$, that can be decomposed into cycles. Then, the procedure extracts cycles from this subgraph that can be used to construct a feasible solution to the LWSND.

We create $\mathcal{G}$ by first adding to $\bar{\mathcal{A}}$ all service arcs $(l_t, m_{t'}) \in \mathcal{S}$ such that either $\sum_{k \in \mathcal{K}} \tilde{x}^{k\tau}_{l_t m_{t'} l t'} > 0$ for some $k \in \mathcal{K}, l \in \Lambda^+, l \in \Lambda$, or they belong to a cycle used in the AP(LWSND) solution (i.e., $\tilde{z}^{\tau}_{ll'} = 1$). We also add all holding arcs to $\bar{\mathcal{A}}$. We then solve an optimization problem to add service arcs in $\mathcal{A} \setminus \bar{\mathcal{A}}$ to $\bar{\mathcal{A}}$ to ensure that $\mathcal{G}$ can be decomposed into cycles in such a way that each service arc appears in at most one cycle but holding arcs may appear in multiple cycles, as doing so maximizes the number of cycles that can be extracted from $\bar{\mathcal{G}}$. The objective of this optimization problem is to minimize the total cost of the arcs added with respect to the cost coefficient f_{lm}. We note that we formulate and solve this optimization problem as a minimum cost, maximum flow problem (Ahuja et al., 1994). See Vu et al. (2013) for details of how a similar procedure was done for a network design problem with Eulerian-type constraints.

After creating $\bar{\mathcal{G}}$, we next extract a set of cycles, $\tilde{\theta}$, from this network that are guaranteed to satisfy Constraint (5), which ensure that each service is executed at most once. Then, we formulate and solve LWSND($\tilde{\theta}$), which is the LWSND restricted to the cycles in the set $\tilde{\theta}$. We set a time limit on the solution of this mixed integer program of t_{MIP} seconds. We note that this procedure is guaranteed to find a feasible solution to the LWSND in each iteration.

5.4 Intensification and Diversification

Metaheuristics (Glover and Laguna, 1997) often include intensification procedures, wherein a promising region of the solution space is explored deeply, and diversification procedures, wherein the search is directed towards regions of the solution space that have not yet been thoroughly searched. As these procedures often enable the search to find high-quality solutions in limited times, we have included them in our solution approach. We first present our intensification procedure and then the diversification procedure.

For intensification, and similar to how we create a feasible solution to LWSND from a solution to AP(LWSND), we introduce an exact optimization step into the search, wherein LWSND($\tilde{\theta}$) is solved with a commercial mixed integer programming solver. Like above, we again set a time limit on the solution of this MIP of t_{MIP}. Our intensification procedure differs from how we create a solution to LWSND from one to AP(LWSND) in how the set of cycles, $\tilde{\theta}$ is created. Whereas previously the set $\tilde{\theta}$ was derived from the services that appear in the most recent solution to the AP(LWSND), our intensification procedure derives $\tilde{\theta}$ from the services that appear in the last q solutions to the AP(LWSND). The rest of the procedure is as described above. The intensification pro-

cedure is executed when an improved solution has not been found after a predefined number of iterations.

For diversification, because it is the cycles and outsourced services in the solution to AP(LWSND) at a given iteration that dictate the structure of the resulting solution to LWSND, we periodically modify the objective function of AP(LWSND) to avoid frequently used cycles and outsourced services. To do so, we collect the number of times, fre_τ, each cycle τ appears in a solution to AP(LWSND). Similarly, we also collect the number of times $fre_{l_t m_{t'}}$ each outsourced service arc $(l_t, m_{t'})$ appears in a solution to AP(LWSND). Then, if the diversification condition is met for each cycle τ and outsourced service arc $(l_t, m_{t'})$ in the current solution of the approximated problem AP(LWSND), we set the linearization factor $\rho^{k\tau}_{l_t m_{t'} ll'}(t)$ to $\rho^{k\tau}_{l_t m_{t'} ll'}(t)(1 + \epsilon * fre_\tau)$ and $\phi^{ke}_{l_t m_{t'}}(t) = \phi^{ke}_{l_t m_{t'}}(t)(1 + \epsilon * fre_{l_t m_{t'}})$ where ϵ is an algorithm parameter.

We continue to update $\rho(t)$ and $\phi(t)$ in this manner for a fixed number of iterations that is dictated by the algorithm parameter I^{diver}_{max}. Lastly, it is possible for two consecutive solutions of AP(LWSND) to be the same, in which case the slope-scaling procedure will terminate. To continue the execution of slope-scaling when this occurs, we add a penalty value P to the linearization factors as seen in equations (21) and (22). In our experiments, P is set to the objective value of the current solution to AP(LWSND).

$$\rho^{k\tau}_{l_t m_{t'} ll'}(t+1) = \begin{cases} \frac{P + H_{ll'} + F_{l'} + \sum_{(l_t, m_{t'}) \in S \cap \tau} f_{lm}}{\sum_{(l_t, m_{t'}) \in \tau \cap S} \sum_{k \in \mathcal{K}} \sum_{(l_t, m_{t'}) \in \tau \cap S} \sum_{k \in \mathcal{K}} \tilde{x}^{k\tau}_{l_t m_{t'} ll'}} & \text{if } \sum_{(l_t, m_{t'}) \in \tau \cap S} \sum_{k \in \mathcal{K}} \tilde{x}^{k\tau}_{l_t m_{t'} ll'} > 0 \\ \rho^{k\tau}_{l_t m_{t'} ll'}(t) & \text{otherwise} \end{cases} \tag{21}$$

$$\phi^{ke}_{l_t m_{t'}}(t+1) = \begin{cases} \frac{P + f^e_{lm}}{\sum_{k \in \mathcal{K}} \tilde{x}^{ke}_{l_t m_{t'}}} & \text{if } \sum_{k \in \mathcal{K}} \tilde{x}^{ke}_{l_t m_{t'}} > 0 \\ \phi^{ke}_{l_t m_{t'}}(t) & \text{otherwise} \end{cases} \tag{22}$$

6 Computational Results

We next report on an extensive computational study. First, we validate the use of the LWSND. Specifically, we compare the resource acquisition and allocation and transportation decisions prescribed by the model proposed in this paper, with plans prescribed by other techniques. We also study how jointly making these strategic and tactical decisions impacts the plans prescribed. Second, having shown the value in using the LWSND, we study (computationally) the effectiveness of the solution approach proposed, which we refer to as SSCG(LWSND) (Slope Scaling with Column Generation for the the Location

with Service Network Design Problem), at finding high-quality solutions. We also examine which of its features contribute to its effectiveness. All experiments were performed on a cluster of computers that have 2 Intel Xeon 2.6 GHz processors and 48 GB of RAM and were running Scientific Linux 6.1. We used CPLEX 12.4 to solve linear and mixed integer programs. We next describe the instances used in the computational study.

6.1 Problem Instances and Parameters

Our experiments are based on 35 instances (previously used in Crainic et al. (2014) and Andersen et al. (2011)) that were inspired by a rail-based case study. We also derive 14 smaller instances from these 35 in order to benchmark the performance of SSCG(LWSND) against a MIP solver. These 35 instances can be broken down into 7 classes, each of which contains 5 instances. We present details regarding the 35 instances (by class) in Table 1; we label these instance classes 6-12 to maintain consistency with Andersen et al. (2011). We present details regarding the 14 smaller instances in Table 2. We generated the smaller instances by truncating the planning horizons of the original, rail-based instances. We note that by truncating the planning horizon we were able to enumerate the set of potential resource cycles, θ.

Instance class	Terminals	Services	Periods	Service+Holding arcs	Commodities
6	5	15	40	600+200	200
7	5	15	50	750+250	400
8	7	30	30	900+210	200
9	7	30	30	900+210	400
10	7	30	50	1500+350	300
11	10	40	30	1200+300	200
12	10	50	30	1500+350	100

Table 1: Rail-based instances adapted from Andersen et al. (2011).

Instances	Terminals	Services	Commodities	Service + Holding Arcs	
				20 periods	25 periods
1-4	5	15	200	300+100	375+125
5-6	7	30	400	600+140	750+175
7-8	7	30	200	600+140	750+175
9-10	7	30	300	600+140	750+175
11-12	10	40	200	800+200	900+250
13-14	10	50	100	1000+200	1250+250

Table 2: Small instances used for calibration and benchmarking against MIP solver.

We next describe the cost structure of the instances used in the computational study. The cost associated with acquiring a new resource ($H_{Al'}, l' \in \Lambda$) is 1,000. The cost associated with repositioning a resource ($H_{ll'}, l \in \Lambda, l' \in \Lambda$) is generated randomly using the function (rand()%5 + 1) * 100 (rand()%5 generates a random number in the interval [0..4]). The cost associated with executing a service with an owned resource, f_{ij}, is set to 50 times the length of the service (as in Crainic et al. (2014)). The analogous cost associated when using a third-party resource, f^e_{ij}, is $4f_{ij}$. To understand the sensitivity of the effectiveness of SSCG(LWSND) to one aspect of the cost structure, we consider various values for the fixed cost F_l associated with using a resource; in some experiments we choose values that favor the use of owned resources and in others we choose values that favor outsourcing.

The solution approach proposed, the SSCG(LWSND), is governed by four parameters. We list those parameters, as well as the values we considered when calibrating the algorithm in Table 3. We calibrated the algorithm by executing it for each of the 16 possible parameter value combinations on the instances described in Table 2. To make the selection, we computed averages of the optimality gap and execution time for each combination over the 14 instances described in Table 2. We found that the difference between the best and worst among the ten best average gaps is about .7%, suggesting that the algorithm is quite robust with respect to the values of these parameters. We list the value chosen for each parameter in Table 3.

Parameter	Description	Values tested	Value selected
I^{diver}_{max}	Number of diversification iterations	3, 5	3
ϵ	The effect of frequency on linearization factor	0.5, 1	1
t_{MIP}	Time limit t_{MIP} for solving a LWSND by the solver	600, 900	600
α	Parameter for cycle's selection	0.01, 0.1	0.1

Table 3: Algorithm parameters and calibration settings

6.2 Evaluating the worth of LWSND

We believe the value in using the LWSND to make strategic resource acquisition and allocation decisions is that the tactical, service network design, component of the model provides an accurate estimate of the impact of these strategic decisions. However, one need not make decisions in this manner. As such, we compare the costs of a solution to the LWSND with two other methods for determining the acquisition and allocation of resources. All the experiments in this section were performed on the instances described in Table 2. To ensure the analysis is based on high-quality solutions, optimization problems (LWSND, SNDRC) that are solved are done so as MPIs with CPLEX 12.4, sometimes with a very long time limit.

The first method is a shipment volume-based heuristic that assigns a fleet of resources of fixed size to terminals, with the number assigned to a terminal proportional to the total volume of shipments originating at that terminal. With this assignment of resources, the tactical transportation planning problem is an instance of the SNDRC (Crainic et al., 2014). We call this overall strategy "POSITION + SNDRC." The second alternative method we evaluate is one that lets the tactical, transportation layer dictate the allocation of resources. Specifically, while we assume a set of resources assigned to each terminal, we initially ignore these bounds and solve an instance of the SNDRC wherein there are (effectively) an unlimited number of resources at each terminal. This problem yields, in a sense, the "optimal" allocation of resources from the service network design perspective. We then, given the actual allocation of resources to terminals, solve an assignment problem to reposition and acquire new resources to meet this optimal allocation. We call this overall strategy "SNDRC + POSITION."

To compare these two planning methods, we used the instances described in Table 2 and consider three values (500, 1,000, and 2,000) for F_l, the cost of using a resource located at terminal l. We note that we assume F_l is invariant with respect to the terminal l. We report in Table 4 the average percentage reduction in total cost when using the LWSND instead of using the POSITION+SNDRC approach. We report similarly-calculated savings in comparison to the SNDRC+POSITION approach in Table 5. We see that using the LWSND provides a clear reduction in costs over both alternative strategies. Given that these instances are relatively small, we believe the savings potential associated with solving the LWSND is substantial for carriers that operate large transportation networks.

F_l	$\frac{val_{LWSND} - val_{POSITION+SNDRC}}{val_{LWSND}}$
500	-0.26%
1000	-0.35%

Table 4: LWSND vs Volume-based heursitic for resource assignment.

F_l	$\frac{val_{LWSND} - val_{SNDRC+POSITION}}{val_{LWSND}}$
1,000	-0.58%
2,000	-0.31%

Table 5: LWSND vs Transportaion-focused heursitic for resource assignment.

A critical question for many carriers is whether they should be asset-heavy (meaning they own many of the resources that they use to transport shipments) or asset-light. We next use the LWSND to try and characterize when a carrier should adopt each strategy. To do so, we consider different cost levels associated with the use of owned resources (parameter F_l) and different levels of demand. "High demand" refers to the instances described in Table 2 whereas "Low demand" refers to those instances, albeit with 25% of the commodities (randomly chosen) removed.

We report in Table 6 the average number of resources the carrier should own, the average number of services that should be supported by a third-party resource, and the

percentage of transportation movements by a resource that do not support a service carrying shipments. Not surprisingly, as the cost associated with using a fixed resource increases, the LWSND indicates that a carrier should decrease its fleet size. We see that this is true for both demand levels. Interestingly, we also see a decrease in empty moves; although this is likely due to the fact that there are simply fewer resources.

F_l	# Owned Resources	# Outsourced Services	% Empty Moves
High demand			
500	29.0	2.1	4.2%
1,000	25.9	5.8	2.3%
2,000	19.8	18.0	0.0%
Low demand			
500	8.9	1.0	4.6%
1,000	7.9	2.2	0.9%
2,000	5.5	6.6	0.0%

Table 6: Characterizing optimal fleet size

Having validated the use of the LWSND on some small instances, we next turn our attention to the ability of SSCG(LWSND) to solve larger-scale instances.

6.3 Benchmarking SSCG(LWSND)

We first benchmark the performance of the SSCG(LWSND) against a commercial MIP solver (CPLEX 12.4) on the instances (described in Table 2) that are small enough (because the planning horizon is truncated) that the set of cycles, θ, can be enumerated in a reasonable amount of time. For these experiments, CPLEX was executed with an optimality tolerance of 1% and a time limit of either one or ten hours. We considered two schedule lengths; 20 and 25 periods. For the 20-period instances, CPLEX was executed for one hour and we report in Table 7 the optimality gap associated with the primal solution produced and the dual bound derived by CPLEX at termination. For the 25-period instances, we executed CPLEX for ten hours and report the resulting optimality gap after one and ten hours. Numbers that appear in parentheses indicate the number of instances wherein CPLEX was not able to find a primal solution in the time alotted. We also report the optimality gap associated with the primal solution produced by SSCG(LWSND) after one hour as measured against the dual bound produced by CPLEX at termination (one hour for 20-period instances and ten hours for 25-period instances).

In these experiments we consider three values for the cost, F_l, of using a resource assigned to terminal l; 500, 1000, and 2000. However, we do assume that it does not vary with respect to the terminal (e.g. $F_l = F_{l'}, \ \forall l, l' \in \Lambda$). We also consider two sizes,

small and *large*, for the number of owned resources. We estimated the *small* size in such a way that we believed either resources would need to be acquired or externally owned resources would need to be used to support executed services. We estimated the *large* level in such a way that we believed owned resources would be enough to support the services executed (although repositioning might still be advisable).

	Size of resource fleet					
	Small			Large		
F_l	500	1,000	2,000	500	1,000	2,000
20 periods - CPLEX 1H gap	2.16	2.54	3.23	4.38	4.79	3.44
20 periods - SSCG(LWSND) 1H gap	3.94	4.12	5.08	5.13	4.61	5.59
25 periods - CPLEX 1H gap	4.95(2)	6.58(1)	7.69(2)	7.90(3)	12.29(2)	12.15(3)
25 periods - CPLEX 10H gap	3.00(1)	3.69	4.57	4.35	4.58	5.35
25 periods - SSCG(LWSND) 1H gap	4.15	5.04	6.25	5.15	5.25	6.30

Table 7: CPLEX and SSCG regarding to small instances.

What we see in the first two rows is that CPLEX outperforms SSCG(LWSND) on the 20-period instances and that the performance of both methods degrades as the number of owned resources increases. However, on average, the dropoff in performance as the number of resources increases is smaller for SSCG(LWSND) than CPLEX. Turning our attention to the 25-period instances we see that SSCG(LWSND) significantly outperforms CPLEX in terms of what each can produce in one hour of execution and SSCG(LWSND) is competitive with what CPLEX can produce in ten hours.

We next turn our attention to the performance of SSCG(LWSND) on the rail-based instances (described in Table 1) that are too large for the full set of cycles, θ, to be enumerated in a reasonable running time. For comparison with SSCG(LWSND), we developed a column generation-based heuristic, which we call CGMIP-H. CGMIP-H uses the column generation portion of SSCG(LWSND) to solve the linear programming relaxation of an instance, keeping the cycles generated, θ. The heuristic then solves the resulting MIP, LWSND(θ), with a commercial MIP solver (CPLEX 12.4).

These experiments also span six scenarios defined by three possible values for F_l (1,000; 3,000; 5,000) and two fleet sizes (*small*, *large*) derived in a manner similar to what was described above. We execute SSCG(LWSND) for 5 hours or 100 iterations, whichever comes first. When executing CGMIP-H we allow CPLEX 5 hours to solve LWSND(θ), and do not limit the amount of time spent in the column generation phase. We report the results of these experiments in Table 8. Again, a number in parentheses indicates CPLEX was unable to produce a primal solution. In these results we see that SSCG(LWSND) is vastly superior to CGMIP-H, sometimes producing solutions that are 10% better than CGMIP-H can. As with the small instances reported on in earlier tables

we again see that the performance of both methods degrades as the number of available resources increases.

	Size of resource fleet					
	Small			Large		
F_l	1,000	3,000	5,000	1,000	3,000	5,000
SSCG optimal gap	5.25%	6.65%	5.40%	6.31%	7.20%	5.67%
CPLEX gap	15.70%(1)	15.25%(2)	10.31%(1)	16.46%(2)	14.64%(2)	8.73%(1)

Table 8: CPLEX and SSCG for medium- and large-size instances.

7 Conclusions and Future Work

We considered a heretofore unstudied planning problem for consolidation carriers; one that links strategic, resource-acquisition and allocation decisions with tactical, service network design-related decisions. In particular, we extended existing service network design models that recognize the need to manage facility-based resources when operating services and routing shipments to also acquire and allocate resources to terminals. In this sense, we switched the focus of the model from tactical to strategic decision-making.

We focused on a setting wherein the only rule that must be observed when managing a resource is that it must ultimately return to the terminal to which it is assigned. However, as the rules governing the management of resource schedules are very dependent on the transportation setting, we proposed a model that encodes these rules in appropriately-constructed cycles and thus is adaptable. For many instances of even modest sizes, enumerating the set of cycles that observe the rules governing what a resource may do is too time-consuming. Thus, we presented a solution approach that both generates cycles that appear in high-quality solutions and creates said solutions. The solution approach combines multiple algorithmic techniques; including column generation, slope scaling, intensification and diversification procedures, and exact optimization.

We concluded the paper with an extensive computational study. As the problem we studied is, in a sense, "new," we first validated its use. We did so by comparing the plans prescribed by the proposed model of the problem with other methods for determining both the acquisition and allocation of resources and the resulting transportation plan. We found that the problem and our model yielded sets of decisions that were less expensive. To ground that study, we performed it on instances that were small enough that a commercial mixed integer programming solver was able to solve to optimality. Then, having established that our model is worth solving we turned our attention to whether the solution approach proposed was effective. We found that it was, as it out-performed both a commercial mixed integer programming solver and another heuristic for the problem.

The model we proposed is based on the premise that a carrier should balance resource acquisition and allocation costs against the costs incurred transporting customer shipments given those resource-related decisions during a single, representative period of time. Given that these strategic decisions are typically in effect for multiple years there are likely many periods of customer shipment demand that should be considered when making these resource acquisition and allocation decisions. As such, the primary focus of our next work is to incorporate uncertainty into the model; potentially through the use of scenarios. And of course, like the model proposed in this paper, such a model will likely require a customized solution approach.

Acknowledgments

While working on this project, the fourth author was doctoral student with Département d'informatique et de recherche opérationnelle, Université de Montréal, and student member of CIRRELT. Partial funding for this project has been provided by the Natural Sciences and Engineering Council of Canada (NSERC), through its Discovery Grant program and by the EMME/2-STAN Royalty Research Funds (Dr. Heinz Spiess). We also gratefully acknowledge the support of Fonds de recherche du Québec through their infrastructure grants and of Calcul Québec and Compute Canada through access to their high-performance computing infrastructure.

References

R. Ahuja, T. Magnanti, and J. Orlin. Network flows: Theory, algorithms, and applications. *Journal of the Operational Research Society*, 45(11):1340–1340, 1994.

J. Andersen, T.G. Crainic, and M. Christiansen. Service network design with asset management: Formulations and comparative analyses. *Transportation Research Part C: Emerging Technologies*, 17(2):197–207, 2009.

J. Andersen, M. Christiansen, T.G. Crainic, and R. Grønhaug. Branch and price for service network design with asset management constraints. *Transportation Science*, 45 (1):33–49, February 2011.

C. Archetti, M. Speranza, and M. Savelsbergh. An optimization-based heuristic for the split delivery vehicle routing problem. *Transportation Science*, 42(1):22–31, 2008.

C. Barnhart, E.L. Johnson, G.L. Nemhauser, M.W.F. Savelsbergh, and P.H. Vance. Branch-and-price: Column generation for solving huge integer programs. *Operations Research*, 46(3):316–329, 1998.

M. Chouman and T.G. Crainic. Cutting-plane matheuristic for service network design with design-balanced requirements. *Transportation Science*, 49(1):99–113, 2015.

I. Contreras and E. Fernandez. General network design: A unified view of combined location and network design problems. *European Journal of Operational Research*, 219 (3):680 – 697, 2012.

T.G. Crainic. Service network design in freight transportation. *European Journal of Operational Research*, 122(2):272–288, 2000.

T.G. Crainic, B. Gendron, and G. Hernu. A slope scaling/lagrangean perturbation heuristic with long-term memory for multicommodity capacitated fixed-charge network design. *Journal of Heuristics*, 10(5):525–545, 2004.

T.G. Crainic, M. Hewitt, M. Toulouse, and D. M. Vu. Service network design with resource constraints. *Transportation Science*, 2014.

R. De Franceschi, M. Fischetti, and P. Toth. A new ILP-based refinement heuristic for vehicle routing problems. *Mathematical Programming B*, 105:471–499, 2006.

G. Desaulniers, J. Desrosiers, and M. Solomon, editors. *Column generation*. Springer-Verlag, New York, 2005.

A. Erera, M. Hewitt, M. Savelsbergh., and Y. Zhang. Improved load plan design through integer programming based local search. *Transportation Science*, 47(3):412–427, 2013.

FMCSA. Summary of hours of service regulations, December 2014. URL `http://www.fmcsa.dot.gov/regulations/hours-service/summary-hours-service-regulation`

F. Glover and M. Laguna. *Tabu Search*. Kluwer Academic Publishers, Norwell, MA, 1997.

M. Hewitt, G. Nemhauser, and M. Savelsbergh. Combining exact and heuristic approaches for the capacitated fixed-charge network flow problem. *INFORMS Journal on Computing*, 22(2):314–325, 2010.

A. Jarrah, E. Johnson, and L. Neubert. Large-scale, less-than-truckload service network design. *Operations Research*, 57(3), 2009.

D. Kim and P. Pardalos. A solution to the fixed charge network flow problem using a dynamic slope scaling procedure. *Operations Research Letters*, 24(4):195–203, 1999.

D. Kim, C. Barnhart, K. Ware, and G. Reinhardt. Multimodal express package delivery: a service network design application. *Transportation Science*, 33(4):391–407, 1999.

M. F. Lai and H. Lo. Ferry service network design: optimal fleet size, routing, and scheduling. *Transportation Research Part A: Policy and Practice*, 38(4):305–328, 2004.

J. S. Lewinski. Turbulence ahead: The coming pilot shortage and how it came to be. The Atlantic, July 2014.

S. Melkote and M. Daskin. Capacitated facility location/network design problems. *European Journal of Operational Research*, 129(3):481 – 495, 2001a.

S. Melkote and M. Daskin. An integrated model of facility location and transportation network design. *Transportation Research Part A: Policy and Practice*, 35(6):515 – 538, 2001b.

M. Pedersen, T.G. Crainic, and O. Madsen. Models and tabu search metaheuristics for service network design with asset-balance requirements. *Transportation Science*, 43(2): 158–177, 2009.

W. B. Powell. A local improvement heuristic for the design of less-than-truckload motor carrier networks. *Transportation Science*, 20(4):246–257, 1986.

J. D. Schulz. Driver shortage issues remain firmly intact based on ata driver turnover data. Logistics Management, April 2015.

K. Smilowitz, A. Atamtürk, and C. Daganzo. Deferred item and vehicle routing within integrated networks. *Transportation Research Part E: Logistics and Transportation Review*, 39(4):305–323, 2003.

B. Tita. Railcar shortage in u.s. pushes up lease rates. The Wall STreet Journal, May 2014.

C. Trudell. Too many cars and not enough trains prompts buffett to expand rail unit. Bloomberg, March 2015.

D. M. Vu, T.G. Crainic, and M. Toulouse. A three-phase matheuristic for the capacitated multi-commodity fixed-cost network design with design-balance constraints. *Journal of Heuristics*, 19(5):757–795, 2013.